UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/990,569	11/21/2001	Aaron S. Mar	004906.P055	2639

8791 7590 03/29/2005

BLAKELY SOKOLOFF TAYLOR & ZAFMAN
12400 WILSHIRE BOULEVARD
SEVENTH FLOOR
LOS ANGELES, CA 90025-1030

EXAMINER	
MARTIN, CIARA A	
ART UNIT	PAPER NUMBER
2157	

DATE MAILED: 03/29/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/990,569	MAR, AARON S.	
	Examiner	Art Unit	
	Ciara Martin	2157	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *Nov 21, 2001*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-44* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-44* is/are rejected.

7)☒ Claim(s) *3, 25, 15, 37* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. This action is responsive to the application filed on November 21, 2001. Claims 1-44 are pending. Claims 1-44 represent a policy change characterization method and apparatus.

Claim Objections

2. Claims 3 and 25 are objected to because of the following informalities: they should depend from claims 1 and 23 respectively. Appropriate correction is required.

3. Claims 15 and 37 are objected to because of the following informalities: the word "or" should be word replaced with the word "of" in lines 3 and 4 respectively. Appropriate correction is required.

Claim Rejections - 35 USC § 102

4. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

5. Claims 1-44 are rejected under 35 U.S.C. 102(b) as being anticipated by Simmonds et al. U.S. Patent No. 5,893,116.

Simmonds teaches the invention as claimed including a method and machine for replicating network resources that are arranged as a hierarchical tree (see abstract and fig. 1).

6. As per claims 1 and 23, Simmonds teaches:

receiving a new policy tree at a network element in a network, wherein the network element stores a current policy tree of classes for quality of service of packets being processed by the network element (see c. 6, ll. 57-65 and c. 9, ll. 10-13; Simmonds discloses a mobile server which stores replicated resources in a network, the network resources are arranged in a directory subtree; the replicated resources are the current tree, the network resources are the new tree; a mobile server is a network element, and a subtree is a tree);

comparing the classes of the current policy tree with the classes of the new policy tree (see c. 7, l. 66 to c. 8 ll. 1-5; Simmonds discloses replicating and propagating changes made on the network resource tree to the replication database; the replication database is the current tree, and replicating and propagating changes involves first comparing); and

selectively deleting classes of the current policy tree based on the comparison of the classes (see c. 7, l. 66 to c. 8 ll. 1-5 and c. 10, ll. 40-49; Simmonds discloses replicating and propagating changes made on the network resource tree to the replication database and selecting objects to delete based on the comparison; the replication database is the current tree, and replicating and propagating changes involves first comparing).

7. As per claims 2 and 24 Simmonds teaches the classes of the current policy tree and the classes of the new policy tree comprise leaf classes and non-leaf classes (see c. 9, ll. 45-45; Simmonds discloses the replication database stores information container objects and leaf objects; an object is a class, the replication database is the current tree and container objects are non-leaf classes).

8. As per claims 3 and 25 Simmonds teaches comparing of the classes of the current policy tree with the classes of the new policy tree comprises:

for the current policy tree and the new policy tree, merging, into a set of classification rules of the leaf classes, classification rules of non-leaf classes that are ancestors of the leaf classes (see c. 9, ll. 60-67; Simmonds discloses leaf objects inherit inheritance rules from the container; inheritance rules are classification rules and containers are non-leaf classes).

identifying a leaf class in the current policy tree as identical to a leaf class in the new policy tree upon determining that the set of classification rules of the leaf class in the current policy tree is equal to the set of classification rules of the leaf class in the new policy tree (see c. 12, ll. 39-52; Simmonds discloses synchronizing leaf objects from a replica if a change has occurred; it is inherent that if a change has not occurred then there is no synchronization and the leaf objects are identical).

identifying a non-leaf class in the current policy tree as identical to a non-leaf class in the new policy tree upon determining that the non-leaf class in the current policy tree and the non-leaf class in the new policy tree share a greatest number of equivalent descendant leaf classes (see c. 12, ll. 39-52; Simmonds discloses synchronizing

resources from a replica if a change has occurred; it is inherent that if a change has not occurred then there is no synchronization and the resources are identical; a non-leaf object is a resource); and

marking the classes of the current policy tree and the new policy tree as added, deleted, modified or unchanged based on the identifying of the identical leaf and non-leaf classes in the current policy tree and new policy tree (see c. 12, ll. 39-58 and table 1 in column 13; Simmonds discloses changes in resources in both the replica and the network resource tree, possible change types include unchanged, changed [or modified] and deleted; changes are marked on the resources [classes]).

9. As per claims 4 and 26 Simmonds teaches selectively deleting classes of the current policy tree comprises deleting a class of the current policy tree upon determining that a set of classification rules of the class of the current policy tree is different than a set of classification rules of a corresponding class of the second policy tree (see c. 13, ll. 46-50 and c. 14, ll. 1-9; Simmonds discloses clash handling of incompatible changes in the replica and master are resolved by overwriting the replica with the master, overwriting can involve deleting objects with different inheritance policies [classification rules]).

10. As per claims 5 and 27 Simmonds teaches each class in the current and new policy tree is positioned at a level in the current and new policy tree and wherein the selectively deleting classes of the current policy tree comprises deleting a leaf class of the current policy tree upon determining that that the leaf class of the current policy tree is not positioned at a same level as a leaf class of the new policy tree (see c. 13, ll. 46-

50 and c. 14, ll. 1-9; Simmonds discloses clash handling of incompatible changes in the replica and master are resolved by overwriting the replica with the master, overwriting can involve deleting an object it does not follow the same tree structure of the master; following the same tree structure involves being at the same level).

11. As per claims 6 and 28 Simmonds teaches selectively deleting classes of the current policy tree comprises selectively deleting at least one leaf class of the current policy tree (see c. 13, ll. 46-50 and c. 14, ll. 1-9; Simmonds discloses clash handling of incompatible changes in the replica and master are resolved by overwriting the replica with the master; the replica has leaf objects).

12. As per claims 7 and 29 Simmonds teaches selectively deleting classes of the current policy tree comprises selectively deleting at least one non-leaf class of the current policy tree (see c. 13, ll. 46-50 and c. 14, ll. 1-9; Simmonds discloses clash handling of incompatible changes in the replica and master are resolved by overwriting the replica with the master; the replica has non-leaf objects).

13. As per claims 8 and 30 Simmonds teaches a class having a parent class further includes all classification rules included in the parent class (see c. 9, ll. 61-67 and c. 10, ll. 1-12; Simmonds discloses leaf objects inherit inheritance rules from the container and containers inherit inheritance rules from its parent container; inheritance rules are classification rules).

14. As per claims 9 and 31 Simmonds teaches each class is positioned at a level in a policy tree and wherein a leaf class of the current policy tree is identical to a leaf class of the new policy tree only if the leaf class of the current policy tree and the leaf class of

the new policy tree are positioned at an equal level (see c. 12, ll. 39-58 and table 1 in column 13; Simmonds discloses changes in resources in both the replica and the network resource tree, possible change types include unchanged; if a leaf is unchanged then it is positioned at the same level on both trees and thus, identical).

15. As per claims 10 and 32 Simmonds teaches each leaf class in the current policy tree and the new policy tree is reciprocally linked to an associated path of non-leaf classes in the current policy tree and new policy tree, respectively, and wherein the selectively deleting the classes of the current policy tree comprises deleting each leaf class in the current policy tree upon determining that the associated path of non-leaf classes in the current policy tree is different from the path of non-leaf classes in the new policy tree for a leaf class (see c. 12, ll. 39-58 and table 1 and c. 14, ll. 1-9; Simmonds discloses changes in resources in both the replica and the network resource tree, possible change types include missing directories [or parents] (non-leaf classes), and clash handling of incompatible changes in the replica and master are resolved by overwriting the replica with the master; overwriting can involve deleting leaf objects without parents).

16. As per claims 11 and 33 Simmonds teaches each class in the current and new policy tree is positioned at a level in the current and new policy tree, wherein each leaf class in the current policy tree and the new policy tree is reciprocally linked to an associated path of non-leaf classes in the current policy tree and new policy tree, respectively, and wherein the selectively deleting classes of the current policy tree comprises deleting a leaf class of the current policy tree upon determining that the

associated path of non-leaf classes linked to the leaf class of the current policy tree includes a non-leaf class positioned at a different level than a corresponding non-leaf class included in the associated path of non-leaf classes linked to the leaf class of the new policy tree (see c. 12, ll. 39-58 and table 1 and c. 14, ll. 1-9; Simmonds discloses changes in resources in both the replica and the network resource tree, possible change types include missing directories [or parents] (or non-leaf classes), and clash handling of incompatible changes in the replica and master are resolved by overwriting the replica with the master; overwriting can involve deleting leaf objects with different parents at different levels).

17. As per claims 12 and 34 Simmonds teaches selectively deleting classes of the current policy tree comprises deleting a leaf class of the current policy tree upon determining that all ancestors of the leaf class of the current policy tree and corresponding ancestors of the leaf of the new policy tree have fewer identical descendant classes than those had by a class of the current policy tree and a class of the new policy tree positioned at the same level as the ancestors of the leaf class of the current policy tree and the ancestors of the leaf class of the new policy tree (see c. 12, ll. 39-58, table 1 in column 13 and c. 14, ll. 1-9; Simmonds discloses changes in resources in both the replica and the network resource tree, possible change types include changed; if a leaf is changed then it is not identical resulting in a clash and clash handling of incompatible changes in the replica and master are resolved by overwriting the replica with the master; overwriting can involve deleting leaf objects if they are not identical).

18. As per claims 13 and 35 Simmonds teaches:

comparing a first policy tree of nodes with a second policy tree of nodes, wherein each node is reciprocally associated with a class of network packets, each class including a set of rules (see c. 7, l. 66 to c. 8 ll. 1-5 an d c.9, ll. 45-49; Simmonds discloses replicating and propagating changes made on the network resource tree to the replication database; the replication database is the current tree, replicating and propagating changes involves first comparing, the replication database (or tree) stores information on associated objects (or nodes) with inheritance rules).

selectively adding classes of the second policy tree to the first policy tree based on the comparison of the classes (see c. 7, l. 66 to c. 8 ll. 1-5 and c. 13, ll. 27-43; Simmonds discloses replicating and propagating changes made on the network resource tree to the replication database and adding objects (or classes) which exist on the master, but do not exist on the replica; replicating and propagating changes involves first comparing).

19. As per claims 14 and 36 Simmonds teaches selectively adding classes of the second policy tree to the first policy tree comprises adding a node of the second policy tree to the first policy tree upon determining that the set of rules associated with the node of the first policy tree is different that the set of rules associated with the corresponding node of the second policy tree (see c. 13, ll. 46-50 and c. 14, ll. 1-9; Simmonds discloses clash handling of incompatible changes in the replica and master are resolved by overwriting the replica with the master; overwriting can involve adding objects (or nodes) with different inheritance policies).

20. As per claims 15 and 37 Simmonds teaches each node in the first and second policy tree is positioned at a level in the first and second policy tree and wherein the selectively adding classes of the second policy tree to the first policy tree comprises or adding a leaf node of the second policy tree to the first policy tree upon determining that that the leaf node of the first policy tree is not positioned at a same level as a leaf node of the second policy tree (see c. 13, ll. 46-50 and c. 14, ll. 1-9; Simmonds discloses clash handling of incompatible changes in the replica and master are resolved by overwriting the replica with the master; overwriting can involve adding an object (or leaf node) if it doesn't follow the same tree structure of the master; following the same tree structure involves being at the same level).

21. As per claims 16 and 38 Simmonds teaches selectively adding classes of the second policy tree to the first policy tree comprises selectively adding a leaf node of the second policy tree to the first policy tree (see c. 13, ll. 46-50; Simmonds discloses adding objects (or leaf nodes) to the replica which exist on the master, but do not exist on the replica).

22. As per claims 17 and 39 Simmonds teaches selectively adding classes of the second policy tree to the first policy tree comprises selectively adding a non-leaf node of the second policy tree to the first policy tree (see c. 13, ll. 46-50; Simmonds discloses adding objects (or non-leaf nodes) to the replica which exist on the master, but do not exist on the replica).

23. As per claims 18 and 40 Simmonds teaches a class associated with a node having a parent node further includes all rules included in a class associated with the

parent node (see c. 9, ll. 61-67 and c. 10, ll. 1-12; Simmonds discloses leaf objects inherit all the inheritance rules from the container [or parent node] and containers inherit all the inheritance rules from its parent container).

24. As per claims 19 and 41 Simmonds teaches each node is positioned at a level in the first and second policy tree of nodes and wherein a first leaf class is identical to a second leaf class only if a leaf node associated with the first leaf class and a leaf node associated with the second leaf class are positioned at an equal level (see c. 12, ll. 39-58 and table 1 in column 13; Simmonds discloses changes in resources in both the replica and the network resource tree, possible change types include unchanged; if an object [or leaf class] is unchanged then it is positioned at the same level on both trees and, thus, identical).

25. As per claims 20 and 42 Simmonds teaches each leaf node in the first policy tree of nodes and the second policy tree of nodes is reciprocally linked to an associated path of non-leaf nodes in the first policy tree of nodes and second policy tree of nodes, respectively, and wherein the selectively adding classes of the second policy tree to the first policy tree comprises adding each leaf node in the second policy tree of nodes to the first policy tree of nodes upon determining that the associated path of non-leaf nodes in the first policy tree of nodes is different from the path of non-leaf nodes in the second policy tree of nodes for a leaf node (see c. 12, ll. 39-58 and table 1 and c. 14, ll. 1-9; Simmonds discloses changes in resources in both the replica and the network resource tree, possible change types include missing directories [or non-leaf nodes], and clash handling of incompatible changes in the replica and master are resolved by

overwriting the replica with the master; missing non-leaf nodes result in a different path of non-leaf nodes and overwriting can involve adding leaf nodes).

26. As per claims 21 and 43 Simmonds teaches each node is positioned at a level in the first and second policy tree of nodes, wherein each leaf node in the first policy tree of nodes and the second policy tree of nodes is reciprocally linked to an associated path of non-leaf nodes in the first policy tree of nodes and second policy tree of nodes, respectively, and wherein the selectively adding the nodes of the second policy tree to the first policy tree comprises replacing a leaf node in the first policy tree of nodes with a corresponding leaf node in the second policy tree of nodes upon determining that the associated path of non-leaf nodes linked to the leaf node of the first policy tree includes a non-leaf node positioned at a different level than a corresponding non-leaf node included in the associated path of non-leaf nodes linked to the leaf node of the second policy tree (see c. 12, ll. 39-58 and table 1 and c. 14, ll. 1-9; Simmonds discloses changes in resources in both the replica and the network resource tree, possible change types include missing directories [or non-leaf node], and clash handling of incompatible changes in the replica and master are resolved by overwriting the replica with the master; missing non-leaf nodes can result in nodes being at different levels in the tree and overwriting can involve adding leaf objects with different parents on different levels).

27. As per claims 22 and 42 Simmonds teaches selectively adding the nodes of the second policy tree to the first policy tree comprises adding a leaf node in the second policy tree of nodes to the first policy tree of nodes upon determining that all ancestors

of the leaf node of the first policy tree and corresponding ancestors of the leaf node of the second policy tree have fewer identical descendant nodes than those had by a node of the first policy tree and a node of the new policy tree positioned at the same level as the ancestors of the leaf node of the first policy tree and the ancestors of the leaf node of the second policy tree (see c. 12, ll. 39-58, table 1 in column 13 and c. 14, ll. 1-9; Simmonds discloses changes in resources in both the replica and the network resource tree, possible change types include changed; if a leaf is changed then it is not identical resulting in a clash and clash handling of incompatible changes in the replica and master are resolved by overwriting the replica with the master; overwriting can involve adding leaf objects if they are not identical).

Conclusion

28. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. Sonderegger et al. U.S Patent No. 5,692,129, Bodine et al. U.S. Patent No. 5,987,471, Foster, Caxton, A Generalization of AVL Trees, Copyright 1973, Association for Computing Machinery, Inc. and Foster, C. C., Information Storage and Retrieval Using AVL Trees, ACM 20th National Conference/1965.

29. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Ciara Martin whose telephone number is 571-272-7507. The examiner can normally be reached on M-F 6:30- 4:00 with second Fridays off.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Ario Etienne can be reached on 571-272-4001. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

CM
3/17/05

SALEH NAJJAR
PRIMARY EXAMINER